OPEN ACCESS Freely available online

PLoS COMPUTATIONAL BIOLOGY

Spatio-Temporal Credit Assignment in Neuronal Population Learning

Johannes Friedrich, Robert Urbanczik, Walter Senn*

Department of Physiology, University of Bern, Bern, Switzerland

Abstract

In learning from trial and error, animals need to relate behavioral decisions to environmental reinforcement even though it may be difficult to assign credit to a particular decision when outcomes are uncertain or subject to delays. When considering the biophysical basis of learning, the credit-assignment problem is compounded because the behavioral decisions themselves result from the spatio-temporal aggregation of many synaptic releases. We present a model of plasticity induction for reinforcement learning in a population of leaky integrate and fire neurons which is based on a cascade of synaptic memory traces. Each synaptic cascade correlates presynaptic input first with postsynaptic events, next with the behavioral decisions and finally with external reinforcement. For operant conditioning, learning succeeds even when reinforcement is delivered with a delay so large that temporal contiguity between decision and pertinent reward is lost due to intervening decisions which are themselves subject to delayed reinforcement. This shows that the model provides a viable mechanism for temporal credit assignment. Further, learning speeds up with increasing population size, so the plasticity cascade simultaneously addresses the spatial problem of assigning credit to synapses in different population neurons. Simulations on other tasks, such as sequential decision making, serve to contrast the performance of the proposed scheme to that of temporal difference-based learning. We argue that, due to their comparative robustness, synaptic plasticity cascades are attractive basic models of reinforcement learning in the brain.

Citation: Friedrich J, Urbanczik R, Senn W (2011) Spatio-Temporal Credit Assignment in Neuronal Population Learning. PLoS Comput Biol 7(6): e1002092. doi:10.1371/journal.pcbi.1002092

Editor: Boris S. Gutkin, École Normale Supérieure, College de France, CNRS, France

Received November 10, 2010; **Accepted** May 2, 2011; **Published** June 30, 2011

Funding: This work was supported by the Swiss National Science Foundation (SNSF, Sinergia grant CRSIKO-122697) and a grant from the Swiss SystemsX.ch initiative (evaluated by the SNSF). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

* E-mail: senn@pyl.unibe.ch

Introduction

Learning from reinforcement involves widely differing spatial and temporal scales both within the behavioral decision making process itself as well as when relating decisions to outcomes. Since they are adaptive, synapses may be viewed as the elementary decision making entities in the brain. But the presynaptic input of any single synapse will contain only very limited information about the task and, further, the millisecond duration of a synaptic release is much shorter than behaviorally relevant time scales. The behavioral decision results from a spatio-temporal aggregation of synaptic releases which is highly non-linear due to e.g. thresholding in the generation of action potentials. Hence the relationship between any single synaptic release and the behavioral decision is not only tenuous but also non-linear.

In relating behavioral decisions to rewarding or unrewarding outcomes, problems arise which are analogous to the ones encountered when relating synaptic releases to decisions. In the "spatial" domain: The state of the world is only partially observable, and hence, what appears to be one and the same decision may sometimes be rewarded and sometimes not. Also, in social interactions, reward may depend on the decisions of other players. In the temporal domain: Whether a decision was appropriate or not may not be immediately obvious and reward may even change with time. Proverbially, short term gain may lead to long term pain (and vice versa).

Hence the spatio-temporal credit assignment problem arises: How can a synapse adapt given that reward delivery is delayed and also depends on the releases of many other synapses as well as on external factors? As one basic mechanism for addressing the temporal problem, theories of reinforcement learning use the eligibility trace, a quantity, decaying exponentially in time, which memorizes the elementary decision up to the time when information about reward becomes available to trigger the persistent adaptive change [1]. Here we point out that a cascade of such synaptic memory traces can in fact provide an integrated solution to the spatio-temporal credit assignment problem by remodulating the presynaptic signal in view of information arising at different stages of the behavioral decision making.

Evidence for synaptic eligibility traces comes from experiments on spike timing dependent plasticity (STDP) where a synaptic release leads to longterm potentiation (LTP) if the neuron emits an action potential shortly thereafter [2,3]. Importantly, the length of the LTP-induction time window (some 15 ms) is on the order of the membrane time constant (τ_M), i.e. it reflects the time during which the synaptic release has influence on somatic action potential generation. The release itself lasts only for some 2 ms, so this form of LTP is most easily accounted for by assuming a local synaptic quantity E_1 providing, just like an eligibility trace, a memory of the release which decays with time constant τ_M. When an action potential is generated, E_1 is read-out to determine a quantity E_2 which, in the simplest interpretation of the STDP

Author Summary

The key mechanisms supporting memory and learning in the brain rely on changing the strength of synapses which control the transmission of information between neurons. But how are appropriate changes determined when animals learn from trial and error? Information on success or failure is likely signaled to synapses by neurotransmitters like dopamine. But interpreting this reward signal is difficult because the number of synaptic transmissions occurring during behavioral decision making is huge and each transmission may have contributed differently to the decision, or perhaps not at all. Extrapolating from experimental evidence on synaptic plasticity, we suggest a computational model where each synapse collects information about its contributions to the decision process by means of a cascade of transient memory traces. The final trace then remodulates the reward signal when the persistent change of the synaptic strength is triggered. Simulation results show that with the suggested synaptic plasticity rule a simple neural network can learn even difficult tasks by trial and error, e.g., when the decision - reward sequence is scrambled due to large delays in reward delivery.

findings, gives the change (Δw) of the synaptic strength [4]. Simply equating E_2 with Δw, however, may be hasty because many repeated pre/post pairings are required in the STDP-protocol to induce a noticeable change. So it seems more reasonable to view E_2 as a second synaptic eligibility trace, keeping a running record of recent pre/post pairings to modulate synaptic strength, perhaps even in a non-linear manner.

As has been widely noted [5–11], one can connect the STDP-findings with reinforcement learning by assuming that the transcription of the second eligibility trace E_2 into the synaptic change Δw is modulated by neurotransmitters like dopamine which provide feedback about external reward (Fig. 1A). Such plasticity rules address the spatial credit assignment problem for synapses sharing a postsynaptic neuron since E_2 captures the relevant correlations between a given synaptic release and the releases of other synapses when they contribute to postsynaptic firing in the neuron. But E_2 does not take into account the interaction in decision making between synapses which have different postsynaptic neurons. For temporal credit assignment, the memory length of E_2 must correspond to the delay between a synaptic release and the delivery of pertinent reward feedback. This delay consists of the time τ_D needed to reach a behavioral decision and the time τ_R for this decision to be rewarded. A value on the order of 1 s seems reasonable for τ_D, but τ_R can easily be much longer, as in a game where multiple decisions are needed to reach a rewarding state. In this case, E_2 simply averages pre/post pairing over multiple decisions even if the firing of the particular neuron was important only for some of the decisions.

Here we propose extending the eligibility trace cascade by a further trace E_3 which takes into account the behavioral decision making process (Fig. 1B). Now the time constant of E_2 is simply τ_D, since E_2 only needs to capture pre/post pairings upto the time when a decision is reached. The decision triggers a transcription of E_2 into E_3 which is modulated by a feedback signal from the decision making circuitry and a signal derived from the firings of the postsynaptic neuron during the decision period. So while E_2 only captures the pre/post correlations, E_3 additionally captures the post/decision correlations. The time constant of E_3 is τ_R, and when reward feedback does become available, the reward together with E_3 determines the synaptic change Δw.

In Text S1 we show that, for a population of spiking neurons feeding into a decision making circuitry (Fig. 1C), such a synaptic cascade can be mathematically derived by calculating the gradient of the expected reward. The resulting gradient ascent rule, however, has a few biologically undesirable aspects. For instance, it requires that E_2 averages pre/post correlations over each decision period. Synapses, however, are unlikely to know when decision periods start and end. For biological realism, we present a modified rule in the main text, where e.g. the averaging over the

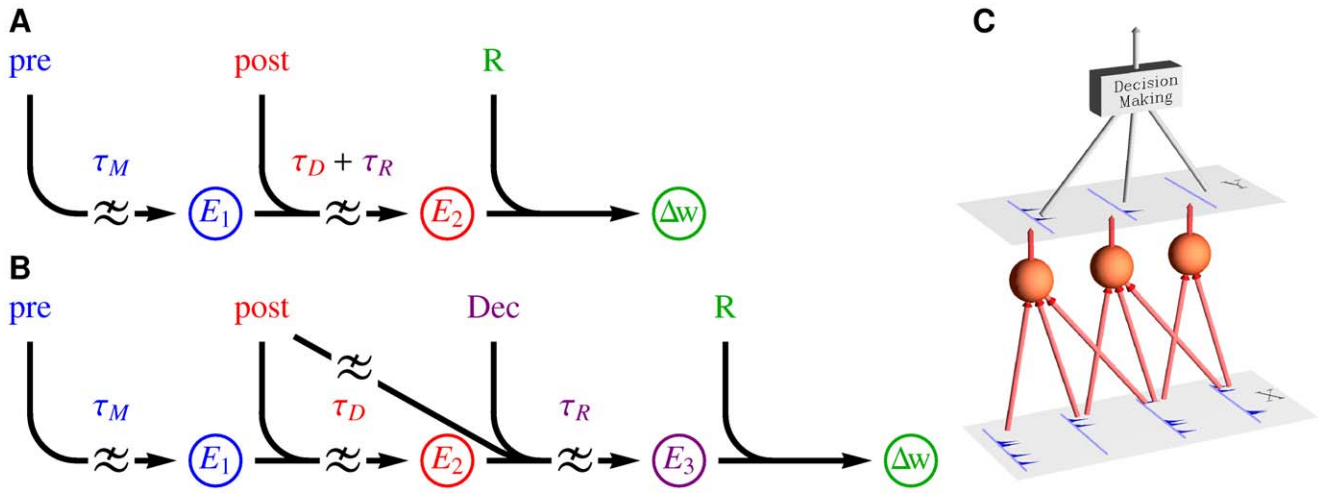

Figure 1. Plasticity cascades and decision making. (A) Synaptic plasticity cascades for reinforcement learning in the single neuron approach and (B) in the proposed population level approach. The meaning of the symbols is the following. $\mathbf{E_i}$: synaptic eligibility traces, **Δw**: change in synaptic strength, **pre**: synaptic input, **post**: feedback from the postsynaptic neuron, **R**: external reward feedback, **Dec**: feedback about the behavioral decision. The symbol ≈ denotes low pass filtering with the time constant τ given next to the symbol. (C) Sketch of the studied population model for reinforcement learning: A stimulus X is read by a population of neurons yielding a spatio-temporal activity pattern Y which depends on the synaptic strength of the neurons. A decision making circuitry transforms the population response Y into a behavioral decision. The synaptic strength of the neurons should adapt so that population responses lead to behavioral decisions which maximize an external reward signal.
doi:10.1371/journal.pcbi.1002092.g001

decision period is replaced by low pass filtering. Learning in a population of spiking neurons using this synaptic plasticity rule is illustrated by simulation results. These show that learning speeds up with increasing population size and that learning speed degrades gracefully when the delay period between decision and reinforcement is increased. In particular, perfect performance is approached even when in the delay period the network has to make further decisions which themselves give rise to delayed reinforcement.

Eligibility traces memorize information about the decision making upto the time when reinforcement becomes available. In contrast, temporal difference (TD) learning, the other basic approach for temporal credit assignment in reinforcement learning, back-propagates reward to the time of the decision. For this, TD-learning estimates the value of states, or state-decision pairs, where, in the simplest case, a state corresponds to a stimulus. The value itself is the (discounted) expected future reward when being in the state, or when making a particular decision in the state. The value can then serve as an immediately available surrogate for the delayed reward signal. During Pavlovian learning, a backward shift in time is observed for the appetitive reaction from the delayed unconditioned stimulus to the conditioned stimulus, and the shift is found as well in the activity of midbrain dopaminergic neurons. The backward shift also occurs in the value estimation error computed by a TD-algorithm modeling the conditioning task, when a state of the algorithm corresponds to the time elapsed since the presentation of the conditioning stimulus [12]. Further to this observation, there has been a surge of interest in modeling dopaminergic activity in terms of TD-learning concepts, as reviewed in [13].

Temporal difference algorithms are based on the assumption that the information available for decision making is rich enough to make the learning problem Markovian. This means that the future is independent of past events, given the current state accessible to the TD-learner. In contrast, eligibility trace based approaches such as our population learning do not require such a completeness of available information. Hence, we present simulation results comparing the performance of the proposed approach to that of TD-learning on tasks, where the Markovian assumption may be violated.

Results

The model

We consider a population of leaky integrate and fire neurons driven by a common presynaptic stimulus and read-out by a decision making circuitry. To facilitate exploration both the population neurons and the decision making are stochastic. As in forced choice tasks, the decision circuitry determines a behavioral choice D at the end of stimulus presentation, based on its monitoring of the population activity for the duration of the stimulus. We focus on binary decision making and denote the two possible behavioral choices by $D=\pm 1$. Immediately, or at some later point in time, a behavioral decision may influence whether reward is delivered to the system, but the decision may also impact the environment, i.e. influence the sequence of stimuli presented to the population neurons. Due to the last point, our framework goes beyond operant conditioning and also includes sequential decision tasks.

For the decision making circuitry itself, we use a very simple model, assuming that it only considers the number of population neurons which fire in response to the stimulus: For low population activity the likely decision is $D=-1$, but the probability of generating the decision $D=1$ increases with the number of neurons that respond by spiking to the stimulus. Given this decision making circuitry, we present a plasticity rule for the synapses of the population neurons, which enables the system to optimize the received reward.

In presenting the plasticity rule we focus on one synapse, with synaptic strength w, of one of the population neurons. (In the simulations, of course, the rule is applied to all synapses of all population neurons.) Let x_t be the set of spike times representing the presynaptic spike train impinging on the synapse upto time t. A presynaptic spike at some time $s_{\mathrm{pre}}\in x_t$ leads to a brief synaptic release with a time constant τ_{s} on the order of a millisecond. The postsynaptic effect of the release will however linger for a while, decaying only with the membrane time constant τ_M which is in the 10 ms range. The first synaptic eligibility trace E_1 bridges the gap between the two time scales by low pass filtering (Fig. 2, column 1). It evolves as:

$$\tau_M \dot{E}_1 = -E_1 + \sum_{s_{\mathrm{pre}}\in x_t} \frac{1}{\tau_{\mathrm{s}}} e^{-(t-s_{\mathrm{pre}})/\tau_{\mathrm{s}}}. \qquad (1)$$

Correlations between synaptic and post-synaptic activity are captured by transcribing E_1 into a second trace E_2 of the form

$$\tau_D \dot{E}_2 = -E_2 + E_1(t)\mathrm{post}_1(t), \qquad (2)$$

see Fig. 2, column 2. The postsynaptic modulation function $\mathrm{post}_1(t)$ depends on the postsynaptic spike times and on the time course $u(t)$ of the neuron's membrane potential. Denoting by Y the set of postsynaptic spike times, the specific form we use for $\mathrm{post}_1(t)$ is

$$\mathrm{post}_1(t) = -k\beta e^{\beta u(t)} + \beta \sum_{s_{\mathrm{post}}\in Y} \delta(t-s_{\mathrm{post}}).$$

Here δ is Dirac's delta-function, k and β are parameters given in Methods.

As has been previously shown [14], E_2 is a useful factor in plasticity rules due to the following properties:

- A small synaptic change proportional to E_2 reinforces the observed neuronal response, i.e. it increases the likelihood that the neuron reproduces the observed postsynaptic spike train on a next presentation of the same stimulus.
- Conversely, a small synaptic change proportional to $-E_2$ impedes the observed neuronal response. It encourages responding by a different spike train on a next presentation of the stimulus and thus facilitates exploration.

Thanks to these properties, plasticity rules where synaptic change is driven by the product of E_2 and reward have been widely used in reinforcement learning models [6,15–17]. Due to E_2, the neuronal quantities modulating plasticity in these rules are not just the pre- and post synaptic firing times but also the membrane potential $u(t)$. This further modulatory factor also arises in models matching STDP-experiments which measure plasticity induction by more than two spikes [18].

In our model, the time constant τ_D in Eq. (2) should be matched to the decision time during which stimuli are presented and we use $\tau_D = 500$ ms. Since the match may be imperfect in reality, we denote the actual stimulus duration by the symbol T. To describe the stochastic decision making in this period, we introduce the population activity variable A which is reset each time one

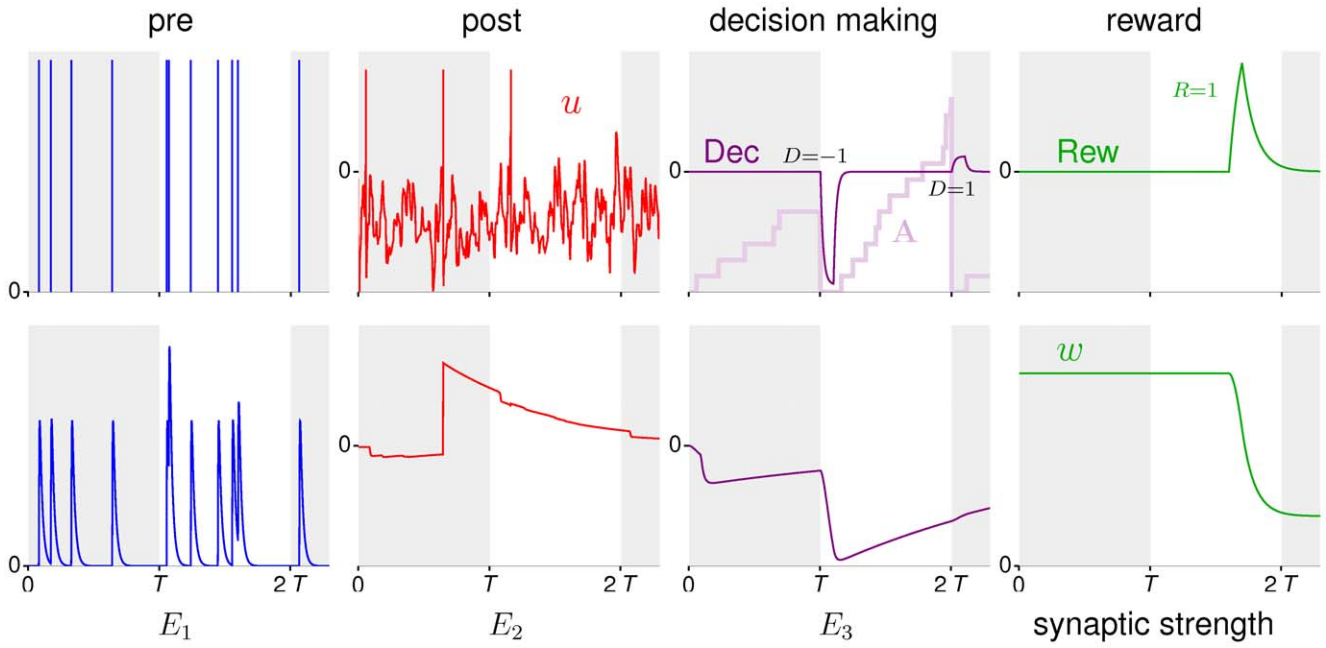

Figure 2. Examples for the modulatory signals and the resulting traces in the plasticity cascade of a synapse. *Top row:* An input stream (stimulus boundaries marked by shading) gives rise to the pre- and the postsynaptic activity shown in the first two panels. The next panel shows the population activity arising from the stimuli as well as the forced choice decisions made at times T and $2T$. Of the total 15 population neurons used in this example simulation, 5 fired during the the first stimulus (i.e. upto time T), for the second stimulus 12 fired. Further, during presentation of the second stimulus external reinforcement generates a reward signal (rightmost panel). *Bottom row:* Each of the stages in the plasticity cascade depends on the trace in the previous stage and the modulatory signals in the top row as indicated by the diagram in Fig. 1B. Mathematically, E_1 is determined by Eq. (1), E_2 by (2), E_3 by (3) and w by (4).
doi:10.1371/journal.pcbi.1002092.g002

decision is made and subsequently increased when a neuron spikes for the first time in response to the next presented stimulus (Fig. 2, column 3). A high (low) value of A at the end of the decision period biases the next behavioral decision towards $D=1$ ($D=-1$). We do not model the temporal accumulation of population activity leading to A explicitly in neural terms, since this could be achieved along the lines previously suggested in [19].

Since the decision circuitry is stochastic, even for a fairly high level of population activity the behavioral decision -1 may be made by chance. In this case, by spiking, a population neuron in fact decreased the likelihood of the behavioral choice which was actually taken, whereas a neuron that stayed silent made the choice more likely. Hence, when the goal is to reinforce a behavioral decision, a sensible strategy is to reinforce a neuronal response when it is aligned with D (firing for $D=1$, not firing for $D=-1$) and to impede it when it is not aligned. To this end, the third eligibility trace E_3 captures the interactions between single neuron activity, population activity and behavioral decision. It evolves as

$$\tau_R \dot{E}_3 = -E_3 + E_2(t)\mathrm{post}_2(t)\mathrm{Dec}(t) \qquad (3)$$

where $\mathrm{Dec}(t)$ is a feedback signal, based on A and D, generated by the decision making circuitry and, further, $\mathrm{post}_2(t)$ is determined by the postsynaptic activity of the neuron. Mathematically, $\mathrm{post}_2(t)$ should reflect how the neuron contributed to the decision and equal ± 1 according to whether or not the neuron fired in response to the decision stimulus. The feedback signal $\mathrm{Dec}(t)$ should consist of pulses generated at the times when a decision D is made. The value of $\mathrm{Dec}(t)$ should have the same sign as the corresponding decision D and be modulated by the population activity A which gave rise to the decision. In particular, the magnitude of the pulse is large when A is close to the stochastic decision threshold, increasing synaptic plasticity in the cases where the decision making is still very explorative.

Since the post-stimulus value of $\mathrm{Dec}(t)$ has the same sign as D, the term $\mathrm{post}_2(t)\mathrm{Dec}(t)$ in Eq. (3) is positive when the neuronal response is aligned with the decision - otherwise it is negative. Because this term remodulates E_2 during the transcription and in view of the above characterization of E_2, the eligibility trace E_3 has the following property:

- A small synaptic change proportional to the post-stimulus value of E_3 reinforces the neurons response when the response is aligned with the behavioral decision but, in the not aligned case, the response is impeded.

Since E_3 encodes the correlations between the releases of the synapse and the behavioral decision, the final stage of the cascade becomes very simple (Fig. 2, column 4). It just remodulates E_3 by reward to yield the synaptic change:

$$\dot{w} = E_3(t)\mathrm{Rew}(t), \qquad (4)$$

Mathematically, the reward function $\mathrm{Rew}(t)$ should be made up of pulses at the times when external reinforcement information becomes available, with the height of each pulse proportional to the reward received at that time.

The above description uses some mathematical idealizations which biologically are not quite realistic. We envisage that the reinforcement and decision feedback is delivered to the synapses by changes in levels of neurotransmitters such as dopamine, acetylcholine or norepinephrine [20–22]. Then, in contrast to the pulses assumed above, the feedback read-out by the synapses

should change only quite slowly. In our simulations, this is addressed by low pass filtering the above feedback pulses when obtaining the signals $\mathrm{Rew}(t)$ and $\mathrm{Dec}(t)$. Further, we assumed above that $\mathrm{post}_2(t)$ in Eq. (3) encodes whether the neuron fired in response to the decision stimulus. But it seems unrealistic, that a population neuron knows when a stimulus starts and ends. In the simulations we use low pass filtering to compute a version of $\mathrm{post}_2(t)$ which just encodes whether the neuron spiked recently, on a time scale given by τ_D (Methods). Such a delayed feedback about postsynaptic activity could realistically be provided by calcium related signaling.

Learning stimulus-response associations with delayed reinforcement

To study the proposed plasticity rule, we first consider an operant conditioning like task, where for each of the stimuli presented to the network, one of the two possible behavioral decisions $D=\pm 1$ is correct. A correct decision is rewarded, whereas an incorrect one is penalized, but in both cases the delivery of reinforcement is delayed for some time. While operant conditioning with delayed reward has been widely considered in the context of temporal discounting [23], here, we are interested in a quite different issue. We do not wish to assume that little of relevance happens in the delay period between the decision and the corresponding reinforcement since this seems artificial in many real life settings. In the task we consider, during the delay period, other decisions need to be made which are themselves again subject to delayed reinforcement (Fig. 3A). Then temporal contiguity between decision and reward is no longer a proxy for causation. So the issue is not how to trade small immediate reward against a larger but later reward, but how to at all learn the association between decision and reward.

In the simulations, a stimulus is represented by a fixed spike pattern made up of 80 Poisson spike trains, each having a duration of $T=500\,\mathrm{ms}$ and a mean firing rate of 6 Hz. To allow for some variability, on each presentation of the stimulus, the spike times in the pattern are jittered by a zero mean Gaussian with a standard deviation of $2\,\mathrm{ms}$. This stimulus representation is used throughout the paper. In the present task, we use 10 stimuli and, for each, one of the two possible decisions is randomly assigned as the correct one. Stimuli are presented in random order and right after the decision on one stimulus has been made, the next stimulus is presented.

Fig. 3B shows learning curves for tasks where there is a fixed delay Δt between each decision and the delivery of the reinforcement pertinent to that decision. Perfect performance is eventually approached, even for the largest value of Δt considered. For this value, $\Delta t=1350\,\mathrm{ms}$, two other decisions are made in the delay period. Learning time increases in a stepwise manner when extending the delay, with a step occurring each time a further intervening decision has to be made in the delay period (Fig. 3B inset).

To demonstrate that the proposed plasticity rule addresses the spatial credit assignment problem as well, we studied learning performance as function of the number N of population neurons. The results in Fig. 3C show that learning speeds up with increasing population size. In a larger population there are more synapses and the speedup indicates that the plasticity rule is capable of recruiting the additional synapses to enhance learning.

To gauge robustness, we used the same synaptic plasticity parameters for all simulations in Panels B and C. In particular τ_R was always set to $1\,\mathrm{s}$ even though the actual delay Δt in reward delivery is varied substantially in Panel B. To further highlight robustness, Fig. 3D shows the performance for different values of τ_R when the actual delay in reward delivery is fixed at $\Delta t=600\,\mathrm{ms}$.

In the above simulations the delay between decision and reward did not change from trial to trial. But the proposed plasticity rule does not rely on this for learning and also works with variable delays. This is shown in Fig. 3E, where a different, randomly chosen, delay Δt was used on each trial.

Two armed bandit with intermittent reward

To achieve near perfect performance in the above operant conditioning task, our network had to learn to make close to deterministic decisions. Here we show that, when appropriate, the architecture can also support stochastic decision making. For this we consider a two armed bandit where one of the two targets delivers a fixed reward of 1 when chosen. The second choice target (which we call intermittent) will deliver a reward of 10 or 0 depending on whether or not the target is baited. Baiting occurs on a variable interval schedule: Once the reward of 10 has been collected, the target becomes un-baited. It stays un-baited for between 6 to 12 time steps (randomly chosen) and is then baited again. Once baited, the target stays in this state until it is chosen. As a consequence, always choosing the intermittent target yields an average reward equal to 1. This does not improve on choosing the fixed reward target and, hence, a better policy is to pick the intermittent target less frequently.

We assume that our network does not have access to the past decisions it has made. Hence on every trial one and the same stimulus is presented to the network (with the same spike pattern statistics as in the previous subsection). The evolution of the average reward collected by the network is shown in Fig. 4A. Due to learning, average reward increases, reaching a value which is within 10% of the reward achievable by the optimal stochastic policy. The probability p_{int} of choosing the intermittent target decreases from 0.5 to around 0.4 as shown in Fig. 4B. This panel also plots the evolution of the value V_{int} of choosing the intermittent target. The value being the expected reward collected from choosing the intermittent target assuming that the policy is to pick this target with a probability of p_{int}.

Asymptotically V_{int} approaches a value around 2.1. So choosing the intermittent target is much more rewarding on average than choosing the fixed target (which has a value of 1). Nevertheless, the intermittent target is chosen less frequently than the fixed target. This amounts to a strong deviation from matching or melioration theory [24] which stipulates that choice frequencies adjust up to the point where the value of the two choices becomes the same - this would lead to $p_{\mathrm{int}}=1$ in the present task. On a task similar to ours, deviations from matching and melioration, favoring a more global optimization of reward, have also been observed in a behavioral experiment with rats [25].

Our plasticity rule, of course, does not explicitly value choices but directly adapts the choice policy to optimize overall reward. This is in contrast to temporal-difference (TD) based approaches to learning, where estimating the value of choices (or, more generally, the value of state-action pairs) is the key part of the learning procedure. Hence it is of interest to compare the above results to those obtainable with TD-learning.

The two most common strategies in TD-learning for making decisions based on the valuation of choices are ε-greedy and softmax. For ε-greedy the choice with the highest estimated value is taken with probability $1-\varepsilon$, where ε is typically a small positive parameter. This does not allow for a fine grained control of the level of stochasticity in the decision making, so we will only consider softmax here. For softmax, a decision i is made with a probability p_i related to its value V_i as $p_i \propto e^{\beta V_i}$. Here the positive

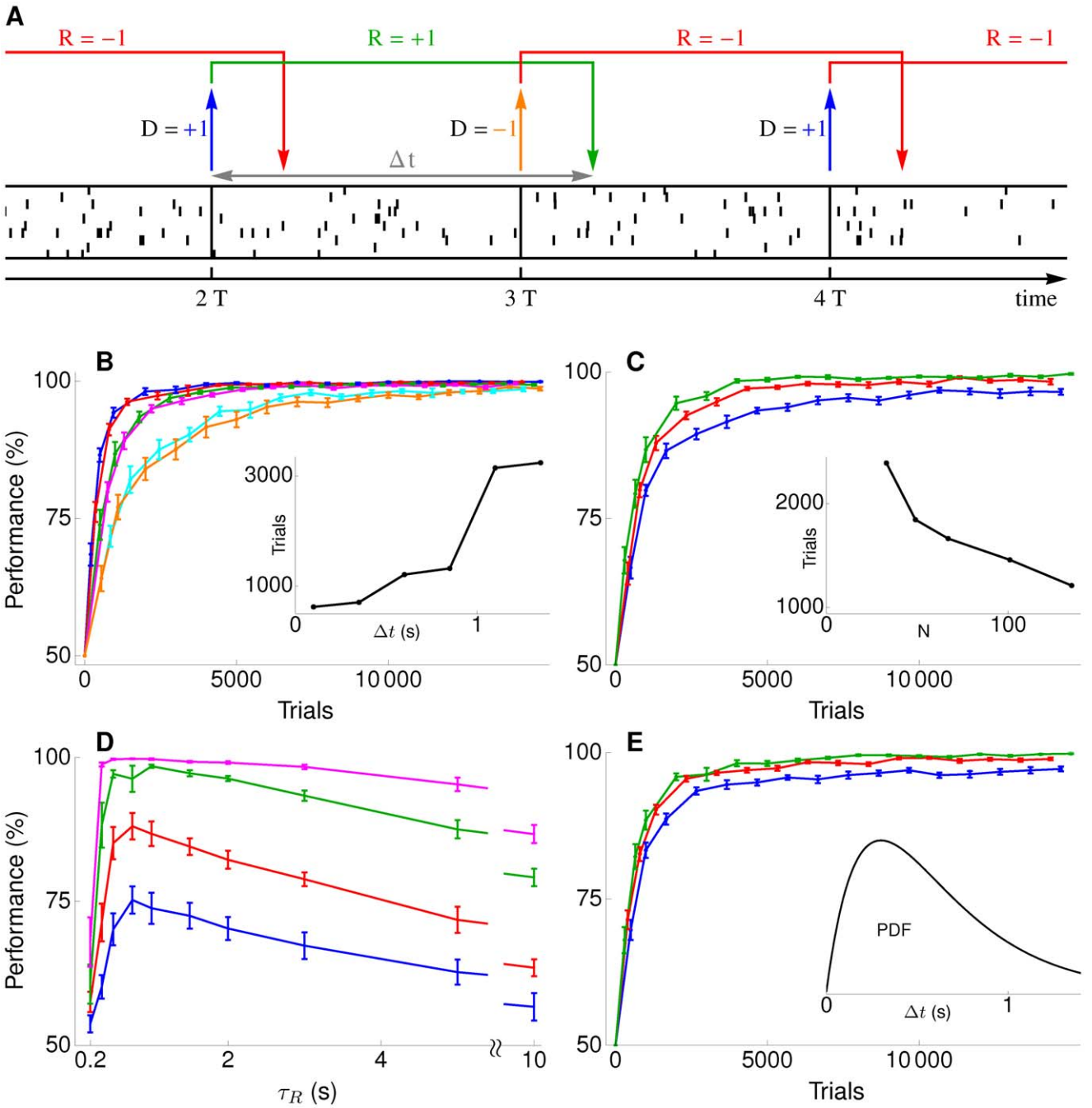

Figure 3. Stimulus-response association with delayed reinforcement. (A) Sketch of the task. After each stimulus (spike pattern of duration T) a decision $D=\pm 1$ is taken and reward $R=\pm 1$ reflecting the correctness of the decision is delivered with a delay Δt. (B) Performance, i.e. the percentage of correct decisions, achieved by a population with $N=135$ neurons for different delays: $\Delta t=100$ ms (blue), $\Delta t=350$ ms (red), $\Delta t=600$ ms (green), $\Delta t=850$ ms (magenta), $\Delta t=1100$ ms (cyan), $\Delta t=1350$ ms (orange). The inset shows the number of pattern presentations required to reach a performance of 90% as function of the delay Δt. (C) Performance vs. number of pattern presentations for a fixed delay $\Delta t=600$ ms but with different population sizes: $N=33$ (blue), $N=67$ (red), $N=135$ (green). Inset: Number of trials needed to reach 90% performance as function of the population size N. (D) Performance as function of the plasticity parameter τ_R representing a guess at the delay between decision and reward. The actual delay was $\Delta t=600$ ms. Performance is plotted after 500 (blue), 1000 (red), 4000 (green) and 15000 (magenta) trials; the population size was $N=135$. (E) Results when the delay Δt is no longer fixed but changes from trial to trial, being randomly chosen from the probability density shown in the inset. The mean delay is $\langle \Delta t \rangle = 600$ ms. Learning parameters and color coding are as in Panel C. In all panels, error bars show 1 SEM of the mean.
doi:10.1371/journal.pcbi.1002092.g003

parameter β, called inverse temperature, modulates the level of stochasticity in the decision making. TD-theory does not give a prescription for choosing β and, hence, we will consider a large range of values for the inverse temperature. The results in panels 4C and 4D plot the asymptotic performance as function of β. Panel 4c shows that the average reward achieved by the TD-learner decreases with increasing β. So best performance is obtained for $\beta=0$, i.e. when the choice valuations estimated during learning are irrelevant. The probability p_{int} of choosing the intermittent target increases with β, Panel 4D. The panel also

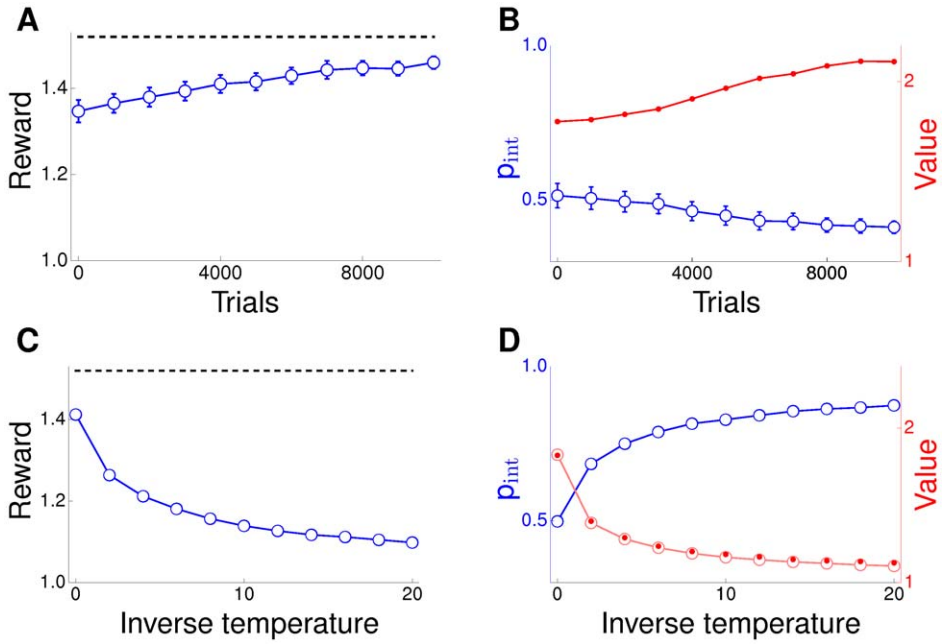

Figure 4. Two armed bandit with intermittent reward. Panels (A) and (B) plot the results for learning with $N = 135$ population neurons and $\tau_R = 3$ s. The evolution of average reward per decision is shown in (A) and compared to the reward achievable by the optimal stochastic policy (dashed line). The latter was determined by Monte Carlo simulation. The probability p_{int} of choosing the intermittent target is shown in (B) as well as the value V_{int}, i.e the average reward obtained when choosing the intermittent target with probability p_{int}. Panels (C) and (D) show the asymptotic performance of TD-learning (reached after 1000 trials) for different values of the inverse temperature β. The red empty circles in panel (D) show the estimate of V_{int} computed by the TD-algorithm. The full red circles give the exact value of V_{int} for the choice probability p_{int} used by the TD-algorithm (blue curve).
doi:10.1371/journal.pcbi.1002092.g004

shows that the estimates of V_{int} computed by the TD-algorithm are in excellent agreement to the true values of V_{int} for the policy characterized by p_{int}. Hence, the TD-learner fails to optimize reward not because the valuation of the decisions is wrong, but it fails because softmax is a poor strategy for transforming valuations into decisions in the present task.

The root cause for the failure of TD-learning is that our decision task is not Markovian. Due to the variable interval schedule, the probability that the intermittent target is baited depends on the previous decisions made by the TD-learner. But as in the simulation on population learning, we have assumed that previous decisions are not memorized and the TD-learner is in the same state in each trial. Hence, even given the state accessible to the TD-learner, past events are nevertheless predictive of future ones because the information about the present encoded in the state is incomplete. This violates the Markovian assumption on which TD-learning theory is based. To rectify this, one needs to assume that decisions are made in view of previous decisions and outcomes. Given that the intermittent target can stay un-baited for a maximum of 12 steps, this requires a TD-learner which memorizes decisions and outcomes (reward/no reward) for the last 12 time steps. Hence, we simulated a TD-learner with the $2^{12} \times 2^{12}$ states needed to represent the task history in sufficient detail to render the decision problem Markovian. We found that the algorithm after learning (with softmax and $\beta = 30$) achieved an average reward of 1.75 ± 0.01 per decision. The algorithm learned to employ sophisticated policies such as not choosing the intermittent target for 8 time steps after it had delivered reward - but polling it frequently thereafter until the intermittent target again delivered reward. Obviously such policies are beyond the scope of the simple memoryless stochastic decision making considered above.

Sequential decision making

We next studied population learning in a sequential decision making task, where reward delivery is contingent on making a sequence of correct decisions. For this, a simple path finding task on a linear track was used (Fig. 5A). We imagine an owner who is tired of having to take his dog for a walk and wants to teach the animal to exercise all by itself. The dog is put in front of the door (position 1 on the track), can move left or right, and may be rewarded on coming home (position 0). But since the point is to exercise the dog, reward ($R = 1$) is only delivered when the dog has reached position 3 at least once while moving on the track. If the dog comes home early without visiting the required position 3, the learning episode simply ends with neither reward or punishment. The episode ends in the same way if position 5 is ever reached (the dog should not run away).

In an initial simulation, we assumed that decisions have to be made based just on the current position on the track. So the stimuli presented to the population just encode this position (using the same spike pattern statistics as in the previous tasks). Given such stimuli, our population model is faced with a non-Markovian decision problem because, the appropriateness of a decision may depend not just on the current stimulus but also on the stimuli which were previously encountered. For instance, whether one should go left or right in position 1 depends on whether position 3 has been visited already. In fact the learning problem is even more dire. When the basis of decision making is just the current position, complete failure will result for any deterministic policy which must lead to one of the following three outcomes: (i) direct exit from position 1 to 0, (ii) exit at position 5, (iii) an infinite cycle. This is not to say that nothing can be learned. As the result in the bottom row of Fig. 5A shows, it is possible to increase the odds that an episode will end with reward delivery by adapting a stochastic

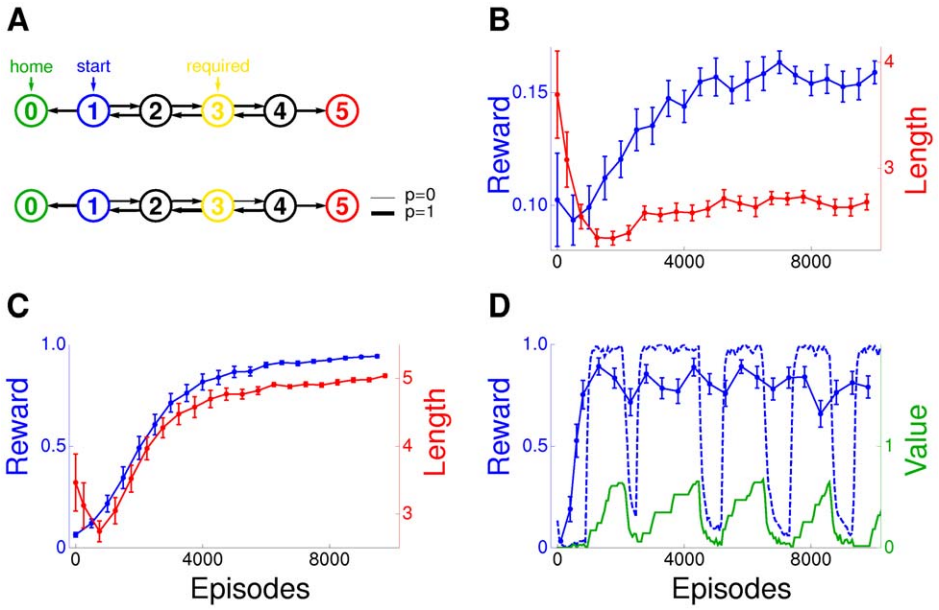

Figure 5. Sequential decision making. (A) Top row, sketch of the path finding task. Bottom row, example stochastic policy learned by the population when decisions are based on just the current position, arrow thickness represents probability of transition. (B) Evolution of the average reward per episode (blue) and the average number of steps per episode (red) for population learning with decisions based on current position. (C) Same as in (B), but for population learning with decisions based on the current and previous position. The above population simulations used $N = 67$ and $\tau_R = 3$ s. (D) TD-learning with decisions based on the current and previous position. Average reward per episode (solid blue curve) and reward per episode in a typical single run (dotted blue). For this run, the green curve shows the evolution of the value assigned by the TD-learner to making a shortcut, i.e. to the state action pair ($_1$2, *left*). Error bars show 1 SEM of the mean.
doi:10.1371/journal.pcbi.1002092.g005

policy. Initially the network was almost equally likely to go left or right in any position but after learning this has changed. In position 3 for instance left is much more likely than right, whereas, in position 2, left is just a little bit more likely than right. After learning, the average number of steps per episode is lower than initially (Fig. 5B, red curve). So in terms of average reward per step taken, there is even more improvement through learning than suggested by the blue curve in Fig. 5B. In the simulations we used $\tau_R = 3$ s. This is somewhat longer than the minimal time of 2.5 s (5 steps of $T = 500$ ms duration) needed from position 1 to reward delivery.

Thanks to working memory, a real dog is of course entirely capable to collect reward by simply running from position 1 to 3 and then back to 0. So for describing the behavior of an animal with a highly developed nervous system, the above model is woefully inadequate. Nevertheless, it may usefully account for behavior in the presence of working memory impairments. To allow for working memory, in a next set of simulations we switched to stimuli encoding not just the current but also the immediately preceeding position on the track. Of the 80 spike trains in a stimulus presented to the network, 50 were used to encode the current and 30 to encode the preceeding position (Methods). Now, learning with the proposed plasticity rule converges towards perfect performance with the reward per episode approaching 1 and the number of decision steps per episode approaching 5 (Fig. 5C).

It is worthwhile noting, that even with a working memory reaching one step back, the decision task is non-Markovian: For instance, knowing that coming from 2 we are now in position 1 does not allow us to tell whether moving left leads to reward. For this we would need to know if we have been in position 3, say, two steps back. Technically, when remembering the sequence of past positions, the memory depth required to make the decision problem Markovian is infinite because any finite memory can be exhausted by cycling many times between positions 1 and 2. The non-Markovian nature of the task is highlighted by Fig. 5D, which shows simulation result for TD-learning. The specific algorithm used is SARSA with ε-greedy decision making (see [1] and Methods). Similarly to Fig. 5C, we assumed that the states upon which the TD-learner bases decisions represents the current and the immediately preceeding position on the track. The solid blue curve in Fig. 5D, computed by averaging performance over multiple runs of the algorithm, demonstrates that TD-learning does not converge towards perfect performance. The dotted blue curve, giving results for a typical single run, shows that in fact TD-learning leads to large irregular oscillations in performance, which are averaged away in the solid curve. While optimal performance is approached initially in the single run, the algorithm is not stable and at some point performance breaks down, initiating a new cycle in the oscillation.

To understand the instability in more detail, we denote the states of the TD-learner by notation such as $_2$1, meaning that coming from 2 the current position is 1. The TD-learner assigns values to state-decision pairs, which we write as e.g. ($_2$1, *left*), by estimating discounted future reward. Now consider the single run of the TD-learner (dotted blue curve, Fig. 5D) after some 1500 episodes. The strategy then is close to optimal, so in most episodes when we are in state $_2$1, i.e. on the inbound leg of the tour, position 3 will have previously been visited. Then *left* in $_2$1 leads to immediate reward delivery, so the state-action pair ($_2$1, left) has a high value. Next assume that we are on the outbound leg in state $_1$2. Since the policy is close to optimal, in most episodes the next move is *right*, in order to visit position 3. But, due to exploration, the TD-learner will occasionally try the shortcut of going left in state $_1$2, testing the state-action pair ($_1$2, *left*). This leads to state $_2$1 and then most likely to the high value decision *left*, terminating the episode without reward

because the shortcut was taken. But the TD-learner updates the value of the tested state-action pair ($_1$2, *left*) based not on the failure at the very end of the episode but based on the value of the subsequent state-action pair, in this case ($_2$1, *left*). As noted above, the latter pair has high value, so the update increases the value of the shortcut ($_1$2, *left*) even-though the shortcut resulted in failure (green curve in Fig. 5D). This happens most of the times when the shortcut is tested for exploration, leading to further increases in the green curve, upto the point where the value of ($_1$2, *left*) is so high that making a shortcut becomes the dominant policy. This causes the observed breakdown in performance. In summary, a central idea in temporal difference learning is to handle non-immediate reward by back-propagating it in time via the valuations of intermediate state-decision pairs. This is mathematically justified in the Markovian case, but may lead to unexpected results for general sequential decision making tasks.

Discussion

We have presented a model of reinforcement learning in a population of spiking neurons read out by a decision making circuitry where plasticity induction is controlled by a cascade of synaptic memory traces. In each synapse of the population neurons, the presynaptic trace is in stages remodulated by somatic feedback, by feedback about the behavioral decision making and by an external reward signal before being consolidated into a persistent change of the synaptic strength. Simulation results show that this leads to robust learning performance in a variety of reinforcement tasks.

Our model builds on, but goes beyond, the classical STDP findings [2,3,26]. On the neuronal level, we assume that plasticity does not only depend on the timings in a pre- and postsynaptic spike pair but that there is a further modulation by postsynaptic subthreshold activity. Such a modulation also arises when modeling the plasticity findings obtained when the standard STDP-protocol is extended to allow multi spike interactions [18]. For reinforcement learning, plasticity cannot be blind to activity-related downstream information. This matches with experimental observations revealing that the polarity and magnitude of STDP can in fact be regulated by neuromodulators such as dopamine, acetylcholine or noradrenaline which may even revert the sign of the synaptic change [10,21,22], e.g. by entering after the mGluR signaling pathways [27–29]. Some recent research has further highlighted astrocytes as local communication elements which are capable of modulating synaptic plasticity [30,31]. Research on synaptic tagging has revealed the astonishingly large time span during which the consolidation of early-LTP into long lasting synaptic change can be dependent on behavioral reinforcement [32,33]. The present work provides a phenomenological model showing how the multi-stage processes observed in the induction of long-term synaptic plasticity can be bound into a functional whole.

Previous modeling of population learning has already considered the modulation of plasticity by feedback from the decision circuitry [16,34]. However, in these works the cascade was shortcut, with decision and reward feedback interacting directly in the modulation of plasticity. As a consequence the previous plasticity rule was capable of handling delays between decision and reward feedback only when these where very small, namely a fraction of typical stimulus duration. The present rule achieves a far more general solution to the temporal credit assignment problem by using a further stage in the synaptic cascade to decouple decision from reward feedback. Further, the rule is now based directly on optimizing the average reward rate (Text S1) and not just, as previously, a related objective function. This puts the present approach squarely into the field of policy gradient methods [35–37]. Within this field, our main contribution is to show how the spatial credit assignment problem of distributing the learning between the population neurons can be solved in a biophysically plausible way. As the results in the section on learning stimulus-response association demonstrate, our plasticity rule leads to a learning performance which scales well to large population sizes (a more detailed scaling analysis has been given in [34]). This is in contrast to the straightforward policy gradient approach of treating the neurons as independent agents which results in a rapid deterioration of learning performance with increasing population size [16].

Crucially in our population model neurons need to cooperate in order to receive reward and hence during learning a difficult spatial credit assignment problem arises. The appropriateness of any single neuron response cannot be determined without taking the responses of the other neurons into account and hence synapses in different neurons need to co-adapt in optimizing reward. This is in contrast to previous work [38] modeling a biofeedback experiment in monkeys [39] where reward delivery was contingent on the firings of a single target neuron. In the model [38] background activity was high, so that reinforcement could be increased by simply strengthening the synapses of the target neuron without any need for coordinated adaptation by the other neurons in the system.

Some parameters in our plasticity scheme are related to properties of the learning task. For instance the time constant τ_R in the last stage of the cascade represents a guess at the typical delay between decision and reinforcement. Our simulation results indicate that learning is not overly sensitive to the choice of the synaptic parameters (see e.g. Fig. 3D). Nevertheless, learning does of course deteriorate once the mismatch between synaptic and actual task parameters becomes too large. An intriguing possibility for further increasing robustness could be an inhomogeneous population of neurons. After all, a key point in population coding is to provide redundancy [40,41]. This is borne out by findings in [16] where, with increasing population size, decision performance improves but the correlation between single neuron performance and decision decreases. Hence it is of interest to study learning when different population neurons have different synaptic parameters. Then the neurons with parameters best matched to the task at hand, are expected to learn best. Thanks to their resulting correlated activity, they should be able to carry the population decision because the contributions from the badly learning mismatched neurons should be uncorrelated and thus tend to cancel. Unfortunately, meaningfully testing whether neuronal variability increases robustness in this manner, requires the simulation of population sizes which are an order of magnitude larger than what is currently within our computational reach.

With regard to the temporal credit assignment problem, we think it is important to note that delayed interaction between decision making and reward delivery can arise in diverse manners:

i. *Delays in causation*. Sometimes it just takes a while till the effect of decisions and actions becomes apparent - as when taking a pill against headache.

ii. *Incomplete information*. The stimulus on which the decision is based does not encode all of the decision relevant information. Then previous stimuli and decisions can be of importance to the current decision because they induce a bias on the missing information. A case in point is the two armed bandit task, where previous decisions influence the odds that the intermittent target is baited. If, in contrast, the decision

stimulus where to encode whether or not the intermittent target is baited, optimal decision making would be possible based just on the current stimulus.

iii. Moving towards a rewarding state. Appropriate decisions or actions are needed to navigate through a set of intermediate non-rewarding states towards a rewarding goal - as when first going to the kitchen, then opening the fridge in order to finally get a beer. In contrast, for the sequential decision making task we considered above, reward is not just contingent on reaching the home state but also on the path taken.

Policy gradient methods work in all of the above settings. Of course, missing information can be detrimental to the performance which is achievable at all. But, given this constraint, policy gradient methods will nevertheless optimize the performance. Temporal difference (TD) methods, however, by design handle only problems of type *iii*. In the first two cases TD-learning only applies when the state which serves as basis for the decision making represents the recent task history to the extent that the problem becomes Markovian. Formally, this maps the first two kinds of delays onto the third kind.

Representing the recent task history is what working memory is good for - and working memory is well known to enter into decision making as in delayed match to sample tasks. On the other hand, transforming a non-Markovian into a Markovian decision problem can pose daunting demands on the working memory capacity needed to adequately represent the states in the TD-algorithm. With insufficient working memory the algorithm can fail in two distinct ways. The estimates for the value of some state-action pairs may be wrong (as demonstrated in the sequential decision making task), or, even when the estimates are correct, preferentially choosing the available action with highest estimated value may lead to a suboptimal policy (as in the two armed bandit).

Policy gradient methods such as our population learning rule seem attractive as basic biological models of reinforcement learning because they work in a very general setting. Arguably, this generality is also a drawback. Precisely because the Markovian property is restrictive, exploiting it in the cases where it does apply, can substantially speed up learning. Hence, it is of interest that policy gradient methods can easily be combined with TD-state valuations in the framework of actor-critic methods. This amounts to simply replacing the direct reward signal in the policy gradient plasticity rule with a signal generated by the TD-valuation circuitry. The TD-signal can either be the estimated value of the current state [42] or the value prediction error [15]. Combining policy gradient with TD-valuations in this way, again brings about the Markovian restriction. Hence, if reinforcement learning is to be both robust and fast, issues of metaplasticity arise: How does brain learn how to learn when?

Methods

Population neurons

The model neurons in our population are escape noise neurons [14], i.e. leaky integrate and fire neurons where action potentials are generated with an instantaneous firing rate which depends on the membrane potential. Focusing on one of the population neurons, we denote by $\mathbf{X}$ its input which is a spike pattern made up of M spike trains X_i ($i=1,...,M$). Each X_i is a list of the input spike times in afferent i. We use the symbol Y to refer to the postsynaptic spike train produced by the neuron, Y is also a list of spike times. If the neuron, with synaptic vector $\mathbf{w}$, produces the output Y in response to $\mathbf{X}$, its membrane potential is determined by

$$\tau_M \dot{u} = u_0 - u + \sum_{i=1}^{M} w_i \sum_{s_{pre}\in X_i} \frac{\Theta(t-s_{\mathrm{pre}})}{\tau_s} e^{-(t-s_{\mathrm{pre}})/\tau_S} - \sum_{s_{\mathrm{post}}\in Y} \delta(t-s_{\mathrm{post}}). \quad (5)$$

Here Θ is the unit step function and, further, δ is Dirac's delta function, leading to immediate hyperpolarization after a postsynaptic spike. For the resting potential, denoted above by u_0, we use $u_0=-1$ (arbitrary units). Further, $\tau_M = 10\,\mathrm{ms}$ is used for the membrane time constant and $\tau_S = 1.4\,\mathrm{ms}$ for the synaptic time constant.

By integrating the differential equation, the membrane potential can be written in spike response form as

$$u(t) = u_0 + \sum_{i=1}^{M} w_i \sum_{s_{\mathrm{pre}}\in x_i} \varepsilon(t-s_{\mathrm{pre}}) - \sum_{s_{\mathrm{post}}\in y} \kappa(t-s_{\mathrm{post}}). \quad (6)$$

The postsynaptic kernel $\varepsilon(t)$ and the reset kernel $\kappa(t)$ vanish for $t\le 0$. For $t>0$ they are given by

$$\varepsilon(t) = \frac{1}{\tau_M-\tau_S}\left(e^{-t/\tau_M} - e^{-t/\tau_S}\right) \text{ and } \kappa(t) = \frac{1}{\tau_M} e^{-t/\tau_M}$$

Note that the first eligibility trace E_1 of synapse i can be expressed in terms of the postsynaptic kernel as $E_1(t) = \sum_{s_{\mathrm{pre}}\in X_i} \varepsilon(t-s_{\mathrm{pre}})$.

Action potential generation is controlled by an instantaneous firing rate $\phi(u)$ which increases with the membrane potential. So, at each point t in time, the neuron fires with probability $\phi(u(t))\delta t$ where δt represents an infinitesimal time window (we use $\delta t = 0.2\,\mathrm{ms}$ in the simulations). Our firing rate function is

$$\phi(u) = k e^{\beta u},$$

with $k=0.01$ and $\beta=5$. (In the limit of $\beta\to\infty$ one would recover a deterministic neuron with a spiking threshold $\theta=0$.)

As shown in [14], the probability density, $P_w(Y)$, that the neuron actually produces the output spike train Y in response to the stimulus X during a decision period lasting from $t=0$ to $t=T$ satisfies:

$$\ln P_w(Y) = \sum_{s\in Y} \ln \phi(u(s)) - \int_0^T \mathrm{d}t\, \phi(u(t)). \quad (7)$$

The derivative of $\ln P_w(Y)$ with respect to the strength of synapse i is known as characteristic eligibility in reinforcement learning [35]. For our choice of the firing rate function one obtains

$$\frac{\partial}{\partial w_i} \ln P_w(Y) = \int_0^T \mathrm{d}t\, \mathrm{post}_1(t) E_1(t) \quad (8)$$

where E_1 is the first eligibility trace of the synapse (Eq. 1) and $\mathrm{post}_1(t)$ the postsynaptic signal of the neuron given right below Eq. (2). Note that (8) is similar to our second eligibility trace E_2, see Eq. (2), except that we have replaced the integration over the decision period by low pass filtering with a time constant matched to the stimulus duration. The reason for this is that it seems unbiological to assume that the synapses of the population neurons know when decision periods start and end.

Architecture and decision making

We use the superscript v, running from 1 to N, to index the population neurons. For instance, Y^v is the postsynaptic spike

train produced by neuron v in response to its input spike pattern $\mathbf{X}^v$. As suggested by the notation, the population neurons have different inputs, but their inputs are highly correlated because the neurons are randomly connected to a common input layer which present the stimulus to the network. In particular, we assume that each population neuron synapses onto a site in the input layer with probability $p=0.8$, leading to many shared input spike trains between the neurons.

The population response is read out by the decision making circuitry based on a spike/no-spike code. For notational convenience we introduce the coding function $c(Y^v)$, with $c(Y^v)=-1$, if the there is no spike in the postsynaptic response Y^v, otherwise, if neuron v produce at least one spike in response to the stimulus, $c(Y^v)=1$. In term of this coding function the population activity A being read out by the decision making circuitry can be written as:

$$A(\mathbf{Y})=\frac{1}{\sqrt{N}}\sum_{v=1}^{N} c(Y^v).$$

Using this activity reading, the behavioral decision $D=\pm 1$ is made probabilistically, the likelihood $P(D|A)$ of producing the decision is given by the logistic function

$$P(D|A)=\frac{1}{1+e^{-2DA}}. \qquad (9)$$

Note that due to the $1/\sqrt{N}$ normalization in the definition of A, the magnitude of A can be as large as $\sqrt{N}$. This is why, decisions based on the activity of a large population can be close to deterministic, despite of the noisy decision making circuitry.

Feedback signals and the postsynaptic trace

We start with the reward feedback $\mathrm{Rew}(t)$, modulating synaptic plasticity in Eq. (4). This feedback is encoded by means of a concentration variable c_{Rew}, representing ambient levels of a neurotransmitter, e.g. dopamine. In the absence of reward information, the value of c_{Rew} approaches a homeostatic level c^0_{Rew} with a time constant $\tau_{\mathrm{Rew}}=50\,\mathrm{ms}$. For any point in time s when external reward information R_{s} is available, this reinforcement leads to a change in the production rate of the neurotransmitter. The change is proportional to R_s and lasts for $L_{\mathrm{Rew}}=50\,\mathrm{ms}$. So up to the point in time s' when further reinforcement becomes available, the concentration variable evolves as:

$$\tau_{\mathrm{Rew}}\dot{c}_{\mathrm{Rew}}=-c_{\mathrm{Rew}}+c^0_{\mathrm{Rew}}+R_{\mathrm{s}}\,\Theta(t;s,L_{\mathrm{Rew}}).$$

Here the step function $\Theta(t;s,L_{\mathrm{Rew}})$ equals 1 if $s\le t\le s+L_{\mathrm{Rew}}$, otherwise the function value is zero. The reward feedback read-out at a synapse is determined by the deviation of the current neurotransmitter level $c_{\mathrm{Rew}}(t)$ from its homeostatic value and equals

$$\mathrm{Rew}(t)=\eta\left(c_{\mathrm{Rew}}(t)-c^0_{\mathrm{Rew}}\right).$$

Here the parameter η is the positive learning rate which, for notational convenience, we absorb into the reward signal.

The decision feedback $\mathrm{Dec}(t)$ used in Eq. (3) is encoded in the concentration c_{Dec} of a second neurotransmitter. As for reward feedback, this is achieved by a temporary change in the production rate of the encoding neurotransmitter. For describing c_{Dec}, we assume a stimulus that ended at time nT, evoking the population activity A and behavioral decision D. As shown in Text S1, the value of $\mathrm{Dec}(t)$ should then be determined by the derivative of $\log P(D|A)$ with respect to A and, in view of Eq. (9), this derivative is simply $D-\tanh(A)$. Hence we use

$$\tau_{\mathrm{Dec}}\dot{c}_{\mathrm{Dec}}=-c_{\mathrm{Dec}}+c^0_{\mathrm{Dec}}+(D-\tanh(A))\Theta(t;nT,L_{\mathrm{Dec}})$$

for the temporal evolution of c_{Dec}. Parameter values in the simulations are $\tau_{\mathrm{Dec}}=10\,\mathrm{ms}$ and $L_{\mathrm{Dec}}=50\,\mathrm{ms}$. The above equation holds up to time $(n+1)T$ when the subsequent stimulus presentation ends, at which point the decision variables D and A are replaced by their values for the latter stimulus. The decision feedback $\mathrm{Dec}(t)$ is simply

$$\mathrm{Dec}(t)=c_{\mathrm{Dec}}(t)-c^0_{\mathrm{Dec}}.$$

For the postsynaptic trace $\mathrm{post}_2(t)$ in Eq. (3), we assume a concentration variable C which reflects the spiking of the neuron. Each time there is a postsynaptic spike, C is set to 1; at other times, C decays as $\tau_D\dot{C}=-C$. The value of C should reflect whether or not the neuron spiked in response to the decision stimulus. So, as for the eligibility trace E_2 (see Eq. 2), the relevant time scale is the decision period and this is why the same time constant τ_D is used in both cases. The trace $\mathrm{post}_2(t)$ is obtained as

$$\mathrm{post}_2(t)=sign(C(t)-\vartheta),$$

comparing C to an appropriate threshold ϑ. In the simulation we use $\vartheta=e^{-1.1}$. For the reasoning behind this choice, consider a stimulus ending at time T of duration $T=\tau_D$. The value of $\mathrm{post}_2(t)$ at time T will accurately reflect whether or not the decision stimulus elicited a postsynaptic spike, if we choose $\vartheta=e^{-1}$. But since decision feedback is not instantaneous, the value of $\mathrm{post}_2(t)$ is mainly read-out at times later than T. This is why the smaller value $\vartheta=e^{-1.1}$ seemed a somewhat better choice.

TD-learning

For TD-learning we used the SARSA control algorithm [1] which estimates the values of state-action pairs (s_t,D_t). At each point in time, the value estimates $V(s_t,D_t)$ are updated according to

$$V(s_t,D_t)\leftarrow(1-\alpha)V(s_t,D_t)+\alpha(R_{t+1}+\gamma V(s_{t+1},D_{t+1})).$$

Here α and γ have values between 0 and 1. The parameter α is similar to a learning rate and γ controls the temporal discounting. The above update is done after every transition from a nonterminal state s_t. If s_{t+1} is terminal, then $V(s_{t+1},D_{t+1})$ is defined as zero. When in state s_t, the next action D_t is chosen using either ε-greedy or softmax. In both cases only the values $V(s_t,D)$ pertinent to the current state enter into the decision making.

For memoryless TD-learning in the two armed bandit we used $\alpha=0.01$ and $\gamma=0$. A positive discount factor would not qualitatively change the result. For each of 30 runs per chosen value of β, we simulated 2.000 trials. After 1.000 trials learning had converged and the reported asymptotic quantities are the average over the next 1.000 trials. For learning with memory we used $\alpha=0.1$, $\beta=30$ and $\gamma=0$.

For the sequential decision making task decision selection used ε-greedy with $\varepsilon=0.01$. The discount factor was set to $\gamma=0.9$ and the step-size parameter to $\alpha=0.1$.

With regard to the failure of TD-learning in the sequential decision making task, we note that there are also eligibility trace based versions, SARSA(λ), of the algorithm with the above version corresponding to $\lambda=0$. For $0<\lambda\leq 1$, the value update takes into account not just the next state-action pair but the value of all subsequent state-action pairs. Importantly, for the special case $\lambda=1$ the subsequent values occurring in the update cancel, and the value update is in effect driven directly by the reward signal [1]. So SARSA(1) is just a complicated way of doing basic Monte Carlo estimation of the values. It hence does not assume that the process is Markovian and SARSA(1) does reliably converge towards optimal performance in our task. For $0<\lambda<1$ the procedure interpolates between the two extremes 0 and 1. Consequently the valuation of some state-action pairs (e.g. the shortcut $_1$2, *left*) will then be wrong but the error will be smaller than for $\lambda=0$. If action selection is based on softmax the incorrect valuation will nevertheless be detrimental to decision making. However, this need not always be the case for ε-greedy, due to the thresholding inherent in this decision procedure. In particular, there is a positive critical value for λ (which depends mainly on the discount factor γ) above which the valuation error will no longer affect the decision making. In this parameter regime, SARSA(λ) will reliably learn the optimal policy (upto the exploration determined by ε).

Miscellaneous simulation details

In all the simulations initial values for the synaptic strength were picked from a Gaussian distribution with mean zero and standard deviation equal to 4, independently for each afferent and each neuron.

A learning rate of $\eta=20$ was used in all simulations, except for the 2-armed bandit task where $\eta=0.2$ was used.

In the sequential decision making task with working memory, the population is presented stimuli encoding not just the current but also the immediately preceeding position. For this, each location on the track is assigned to a fixed spike pattern made up of 50 spike trains representing the location in the case that it is the current position and, further, to a second spike pattern with 30 spike trains for the case that it is the immediately preceeding position. The stimulus for the network is then obtained by concatenating the 50 spike trains corresponding to the current position with the 30 spike trains for the preceeding position.

The curves showing the evolution of performance were obtained by calculating an exponentially weighted moving average in each run and then averaging over multiple runs. For the sequential decision making task reward per episode was considered and the smoothing factor in the exponentially weighted moving average was 0.02. In the other task, where performance per trial was considered, the smoothing factor was 0.005. For each run a new set of initial synaptic strength and a new set of stimuli was generated. The number of runs was 20, except in the two armed bandit where we averaged over 40 runs.

Supporting Information

Text S1 We show how the plasticity rule presented in the main text is based on a gradient ascent procedure maximizing the average reward rate.
(PDF)

Acknowledgments

We thank Michael Herzog and Thomas Nevian for helpful discussions on the learning task paradigms and on possible molecular implementations of the synaptic plasticity rule.

Author Contributions

Conceived and designed the experiments: RU WS. Performed the experiments: JF. Analyzed the data: JF. Contributed reagents/materials/analysis tools: JF RU. Wrote the paper: RU WS.

References

1. Sutton RS, Barto AG (1998) Reinforcement Learning: An Introduction. Cambridge, MA: MIT Press.
2. Markram H, Lübke J, Frotscher M, Sakmann B (1997) Regulation of synaptic efficacy by coincidence of postsynaptic APs and EPSPs. Science 275: 213–215.
3. Bi G, Poo M (1998) Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type. J Neurosci 18: 10464–10472.
4. Song S, Abbott LF (2001) Cortical development and remapping through spike timing-dependent plasticity. Neuron 32: 339–350.
5. Baras D, Meir R (2007) Reinforcement learning, spike-time-dependent plasticity, and the BCM rule. Neural Comput 19: 2245–2279.
6. Florian R (2007) Reinforcement learning through modulation of spike-timing-dependent synaptic plasticity. Neural Comput 19: 1468–1502.
7. Izhikevich E (2007) Solving the distal reward problem through linkage of STDP and dopamine signaling. Cereb Cortex 17: 2443–2452.
8. Legenstein R, Naeger C, Maass W (2005) What can a neuron learn with spike-timing-dependent plasticity? Neural Comput 17: 2337–2382.
9. Pawlak V, Kerr JND (2008) Dopamine receptor activation is required for corticostriatal spiketiming-dependent plasticity. J Neurosci 28: 2435–2446.
10. Zhang J, Lau P, Bi G (2009) Gain in sensitivity and loss in temporal contrast of STDP by dopaminergic modulation of hippocampal synapses. Proc Natl Acad Sci USA 106: 13028–13033.
11. Gavornik JP, Shuler MGH, Loewenstein Y, Bear MF, Shouval HZ (2009) Learning reward timing in cortex through reward dependent expression of synaptic plasticity. Proc Natl Acad Sci USA 106: 6826–6831.
12. Schultz W, Dayan P, Montague PR (1997) A neural substrate of prediction and reward. Science 275: 1593–1599.
13. Dayan P, Niv Y (2008) Reinforcement learning: the good, the bad and the ugly. Curr Opin Neurobiol 18: 185–196.
14. Pfister J, Toyoizumi T, Barber D, Gerstner W (2006) Optimal spike-timing-dependent plasticity for precise action potential firing in supervised learning. Neural Comput 18: 1318–1348.
15. Castro D, Volkinshtein S, Meir R (2009) Temporal difference based actor critic learning – convergence and neural implementation. In: Advances in neural information processing systems 21. Cambridge, MA: MIT Press. pp 385–392.
16. Urbanczik R, Senn W (2009) Reinforcement learning in populations of spiking neurons. Nat Neurosci 12: 250–252.
17. Vasilaki E, Frémaux N, Urbanczik R, Senn W, Gerstner W (2009) Spike-based reinforcement learning in continuous state and action space: when policy gradient methods fail. PLoS Comput Biol 5: e1000586.
18. Clopath C, Büsing L, Vasilaki E, Gerstner W (2010) Connectivity reflects coding: a model of voltage-based STDP with homeostasis. Nat Neurosci 13: 344–352.
19. Wang X (2002) Probabilistic decision making by slow reverberation in cortical circuits. Neuron 36: 955–968.
20. Foehring R, Lorenzon N (1999) Neuromodulation, development and synaptic plasticity. Can J Exp Psychol 53: 45–61.
21. Matsuda Y, Marzo A, Otani S (2006) The presence of background dopamine signal converts longterm synaptic depression to potentiation in rat prefrontal cortex. J Neurosci 26: 4803–4810.
22. Seol G, Ziburkus J, Huang S, Song L, Kim I, et al. (2007) Neuromodulators control the polarity of spike-timing-dependent synaptic plasticity. Neuron 55: 919–929.
23. Daw ND, Doya K (2006) The computational neurobiology of learning and reward. Curr Opin Neurobiol 16: 199–204.
24. Mazur J (2002) Learning and Behavior. Upper Saddle River, NJ: Prentice Hall.
25. Baum W, Aparicio C (1999) Optimality And Concurrent Variable-interval Variable-ratio Schedules. J Exp Anal Behav 71: 75–89.
26. Song S, Miller K, Abbott L (2000) Competitive Hebbian learning through spike-timing dependent synaptic plasticity. Nat Neurosci 3: 919–926.
27. Lynch MA (2004) Long-term potentiation and memory. Physiol Rev 84: 87–136.
28. Nevian T, Sakmann B (2006) Spine Ca^{2+} signaling in spike-timing-dependent plasticity. J Neurosci 26: 11001–11013.
29. Abraham W (2008) Metaplasticity: tuning synapses and networks for plasticity. Nat Neurosci 9: 387–399.
30. Volterra A, Meldolesi J (2005) Astrocytes, from brain glue to communication elements: the revolution continues. Nat Rev Neurosci 6: 626–640.
31. Henneberger C, Papouin T, Oliet SH, Rusakov DA (2010) Long-term potentiation depends on release of D-serine from astrocytes. Nature 463: 232–236.

32. Frey S, Frey JU (2008) 'Synaptic tagging' and 'cross-tagging' and related associative reinforcement processes of functional plasticity as the cellular basis for memory formation. Prog Brain Res 169: 117–143.
33. Almaguer-Melian W, Bergado JA, Lopez-Rojas J, Frey S, Frey JU (2010) Differential effects of electrical stimulation patterns, motivational-behavioral stimuli and their order of application on functional plasticity processes within one input in the dentate gyrus of freely moving rats in vivo. Neuroscience 165: 1546–1558.
34. Friedrich J, Urbanczik R, Senn W (2010) Learning spike-based population codes by reward and population feedback. Neural Comput 22: 1698–1717.
35. Williams R (1992) Simple statistical gradient-following algorithms for connectionist reinforcement learning. Mach Learn 8: 229–256.
36. Baxter J, Bartlett P (2001) Infinite-horizon policy-gradient estimation. J Artif Intell Res 15: 319–350.
37. Baxter J, Bartlett P, Weaver L (2001) Experiments with infinite-horizon, policy-gradient estimation. J Artif Intell Res 15: 351–381.
38. Legenstein R, Pecevski D, Maass W (2008) A learning theory for reward-modulated spike-timingdependent plasticity with application to biofeedback. PLoS Comput Biol 4: e1000180.
39. Fetz EE, Baker MA (1973) Operantly conditioned patterns on precentral unit activity and correlated responses in adjacent cells and contralateral muscles. J Neurophysiol 36: 179–204.
40. Pouget A, Dayan P, Zemel R (2000) Information processing with population codes. Nat Rev Neurosci 1: 125–132.
41. Averbeck B, Latham P, Pouget A (2006) Neural correlations, population coding and computation. Nat Rev Neurosci 7: 358–366.
42. Sutton R, McAllester D, Singh S, Mansour Y (2002) Policy gradient methods for reinforcement learning with function approximation. In: Advances in neural information processing systems 12. Cambridge, MA: MIT Press. pp 1057–1063.